Efficient In-flight Transfer Alignment Using Evolutionary Strategy Based Particle Filter Algorithm

Suvendu Chattaraj and Abhik Mukherjee
Department of Computer Science and Technology, Bengal Engineering and Science University, Shibpur, West Bengal, India

Keywords: Transfer Alignment, Nonlinearity, Particle Filter, Evolutionary Strategy, Target Tracking.

Abstract: Large initial misalignment between mother and daughter munitions make transfer alignment system nonlinear, because small angle approximation applicable to the system dynamics does not hold. Further, when the parameters of state transition matrix are based on current measurements, the system becomes time varying. A conventional Kalman filter fails to estimate misalignment in such situations. A particle filter performs satisfactorily, but, the performance suffers when the knowledge about the system is not accurate. Out of particles that get propagated through such improper system dynamics, only a few are retained and used for estimation purpose, due to sample impoverishment problem. In this work, it is claimed that better result can be obtained by employing an evolutionary strategy. Set of support points are generated for each particle by propagating the particle through an array of perturbed system dynamics, and, then by choosing best weight support point as *apriori* estimate from that set. The current work considers design of such evolutionary strategy based particle filter. For the purpose of proving robustness of proposed algorithm, simulation is first carried out on target tracking problem. Then it is applied to in-flight transfer alignment problem and its performance is found to be satisfactory.

1 INTRODUCTION

In the context of delivering guided munitions from a moving platform, Transfer Alignment (TA) refers to the process of determining the orientation of inertial reference axes of daughter munitions with respect to that of the mother platform. Mere copying of mother data to daughter is not enough, as it does not consider the misalignment between the two. Hence, an algorithm is needed to estimate the misalignment. Through some TA algorithm, initial velocity, position and attitude of daughter, imparted by mother vehicle, is determined as accurately as possible prior to ejection, to facilitate further navigation. Various TA techniques have been reviewed in (Chattaraj et al., 2013; Ali and Jiancheng, 2004).

An estimator (like Kalman Filter (KF)) is used in TA problem for estimating system states, which uses the dynamics of the system (based on the Newtons̀ laws of motion) and some noise contaminated measurements. KF can provide optimal estimate for linear systems perturbed by white Gaussian noise, but, in reality, such assumption does not hold, and necessitates use of some variants of KF. Factors such as large initial misalignment angles ($> 5^o$), non white-Gaussian noise models etc., makes TA problem nonlinear and has been discussed and presented in (Dmitriyev et al., 1997).

For a non-linear system, variant of KF, like Extended KF (EKF), Unscented KF (UKF) etc. are used. These approaches handle non-linearity based on the principle of piecewise linear approximation of system model, which, may not give accurate estimation for poorly designed system models, un-modelled non-Gaussian system or measurement noise etc (Wan and Van Der Merwe, 2000; Julier and Uhlmann, 1997).

Another approach to handle non-linearity is Particle Filter (PF) in which, randomly chosen sample points (particles) with associated weights are used to compute posterior density of estimates of system states. PF does not require rigorous noise modelling (as in KF) for system and measurement noise to produce optimal estimate for the system states. However, performance of PF is heavily depended on system modelling and suffers due to loss of samples which represent the solution space (sample impoverishment problem) (Uosaki et al., 2003). Real time system behaviour is unpredictable and cannot be modelled properly by means of system dynamics. For a PF based estimator, if initially generated particles are

Chattaraj S. and Mukherjee A..
Efficient In-flight Transfer Alignment Using Evolutionary Strategy Based Particle Filter Algorithm.
DOI: 10.5220/0005006000050013
In *Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2014), pages 5-13
ISBN: 978-989-758-039-0

propagated through any such improper system dynamics, it results in retention of a few imperfect samples which may affect overall performance.

One solution to the above problem is to employ larger number of particles at the beginning, but, that increases computational complexity of the entire process and thus avoided. Better approach may be to generate few support points for each particle in each iteration representing few more system dynamics, rather than increasing number of particles arbitrarily. This approach may be regarded as an evolutionary strategy (ES) and has been recommended in literature for handling sample impoverishment problem of conventional PF (CPF) (Uosaki et al., 2003; Uosaki et al., 2004). Residual (difference between predicted and measured system states) plays an important role in the estimation. The closer this value is to zero, the closer the system is to being perfectly modelled and hence reliable. Generating support points for particles is essentially strengthening system models by bringing the value of residual close to zero. Such approach is thus termed as Residual Evolutionary Strategy Based Particle Filter (REPF) algorithm in this work.

In the current work, following the design of the proposed algorithm, performance of CPF with 1000 particles (CPF) and REPF are compared for tracking problem. Target tracking is a classical problem in which, some estimator is used to predict the navigation information of manoeuvring target in the next time step. Manoeuvring target emits signal which is received by some sensors placed in the tracker, and the tracker process those signals by applying some estimation algorithm, to estimate navigation information of manoeuvring target. Trackers may use various measurements for this purpose such as position, velocity acceleration or any combination of these. This particular problem has number of applications in real world such as mobile robot localization, mobile (phone) tracking etc. Non-linearity in tracking problem arises mainly due to unpredictable dependencies of measurement noise and system states (Li and Jilkov, 2001) and also on the non-uniform availability of measurements (Li and Jilkov, 2001; Li and Jilkov, 2003).

Application of the proposed algorithm to tracking problem exercises the robustness of the algorithm. It has been shown that, proposed algorithm works at par with that of CPF in situation when the system dynamics is considered perfect and works better in adverse situation. Then similar algorithm is applied to in-flight TA problem. One simulation exercise is conducted with assumption of proper system knowledge. Performance of REPF is found to be at par with other two filters. In the second simulation scenario, a more practical one, the system knowledge is perturbed and performance of CPF with 1000 particles and REPF is compared. The REPF that uses the concept of support points for each particle has been found most suitable to meet accuracy requirement for multiple delivery of munitions, when knowledge about the system is poor.

The presentation is organized as follows. First the TA requirement is discussed in Section 2 where the problems of Kalman filter and conventional particle filter convergence due to presence of nonlinearities in the system matrix is highlighted. Following this, a particle filter design, capable of handling the nonlinearities in system matrix, is proposed in Section 3. The preformance of the designed filter is first tested on tracking problems and the result is given in Section 4. Following satisfactory performance of the filter in case of tracking problem, it is tuned to tackle the nonlinearities of the TA problem and the algorithm is detailed in Section 5. Peformance of the algorithm for TA is analyzed in Section 5.2. Complexity being a major issue in any particle filter design, the complexity analysis of the designed filter is presented in Section 5.3. Scope of further work in this direction is provided in Section 6.

2 BACKGROUND

Misalignment angle results in components of velocity, attitude and position errors between mother and daughter measurements, otherwise, measurements of mother and daughter would have matched. In navigation, these error propagation equations form the basis of the system equations used in the state space formulation of the TA problem (Groves, 1999). Deliberate manoeuvre is used to get appropriate differences in velocity measurement of mother and daughter inertial navigation system (INS). Assumption of small misalignment angle makes the TA problem fit into a KF framework, which is best described by the following state propagation and measurement equations:

$$x_{k+1} = Fx_k + Bu_k + w \tag{1}$$

$$z_k = Hx_k + v \tag{2}$$

where, x_k is the system state vector, F and H are linear functions of system states, w and v are white Gaussian system and measurement noise with covariance Q_k and R_k respectively, and, B is the control matrix. The term Bu_k in this context is the regulator to the system(Bemporad et al., 2002), which computationally corrects the misalignment to zero, without involving any actuator for physical correction of misalignment.

The TA algorithm used in the simulation and real time environment is described in Figure 1. State transition matrix is computed in every iteration from measurement provided by mother INS, which has low noise contamination, as mother INS is costly and more accurate. TA filter uses this externally supplied state transition matrix to predict *apriori* estimate, in the time update phase, which is corrected using measurement to form *aposteriori* estimate once measurement is available for that cycle. This system functions well when mother INS is aided with more accurate measurement, like GPS (Groves, 1999) but suffers in absence of those. Transient nature of external aiding makes the state transition matrix time variant and erroneous, which in turn, affect overall filter performance. Also, factors like large initial misalignment, non white, non Gaussian system / measurement noise, makes system non-linear and a conventional KF fails to provide correct estimation of misalignment in such cases (Julier and Uhlmann, 1997). Such a non-linear, time varying as well as erroneous system necessitate the use of some non-linear filter.

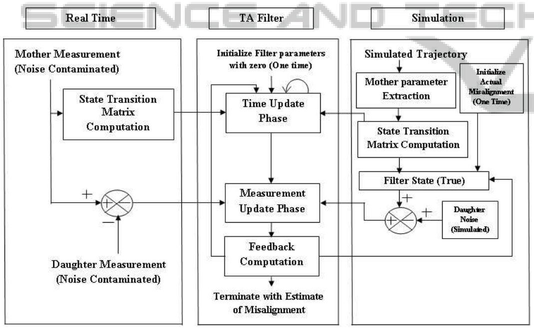

Figure 1: TA in Simulation and real time environment.

For a non-linear system, the state and measurement equations can be expressed as follows:

$$x_{k+1} = f_k(x_k, w_k, u_k) \tag{3}$$

$$z_k = h_k(x_k, v_k) \tag{4}$$

where, f_k and h_k are non-linear functions of system states, with v_k and w_k being noises described as before.

PF is used extensively in non-linear TA problem (Ding et al., 2009), but suffers from sample impoverishment problem which can be solved by resampling. Resampling is costly and its practical implementation is complex, and thus avoided in real time situation like TA problem.

3 PROPOSED FILTER DESIGN

3.1 CPF Algorithm

CPF essentially solves the recursive Bayesian estimation problem by using Monte Carlo approach for a non-linear system. A very good description of PF may be found in (Arulampalam et al., 2002). A CPF tries to estimate the posterior distribution $p(x_{0:k}|z_{1:k})$, i.e. the estimate of x_k based on the available measurements and using a set of randomly chosen samples and associated weights.

A sample importance re-sampling PF (SIR-PF) is considered similar to a genetic algorithm because the steps stated above are similar to the primitive steps of genetic algorithms (GAs) (Kwok et al., 2005) (i.e. initialize population, calculate fitness of individual elements of population and evolve candidate population by applying mutation and crossover). Moreover, the weight calculation phase in SIR-PF bears some similarities with the selection process used in GA as both are probabilistic in nature. Such probabilistic selection is the main cause of sample impoverishment problem just described.

3.2 Proposed REPF Algorithm

Following (Uosaki et al., 2004), in this approach, '*n*' support points $x_k^{(i,j)}$ {i=1,2,...N_s, j=1,2,...n}, are generated per particle $x_{k-1}^{(i)}$ {i=1,2,...N_s)}, sampled from the importance density function $q(x_{k-1}|y_{1:k-1})$, based on the importance density function $q(x_k|x_{k-1}^{(i)} : y_{1:k})$. Weights corresponding to each particle are generated following:

$$\overline{w}_k^{(i,j)} = w_{k-1}^{(i)} \frac{p(y_k|x_k^{(i,j)})p(x_{k-1}^{(i,j)}|x_{k-1}^{(i)})}{q(x_k^{(i,j)}|x_{(k-1)}^{(i)}, y_{1:k})} \tag{5}$$

$i = 1,2,...N_s, j = 1,2,...,n$ The system now has $(n \times N_s)$ particle-weight combination $x_k^{(i,j)}; w_k^{(i,j)}$, and one out of every n support points of a particle is chosen to select N_s most likely particle-weight combinations, from this set. In the context of the TA algorithm design, N_s distinct particles, in the form of state vector x, are considered. Altogether n different state transition matrices (obtained by perturbing elements of F matrix) are used for *apriori* propagation of each such state vector. Choice of most likely system dynamics for the propagation is based on closeness of the corresponding *apriori* estimate state vector to the current measurement vector, and the mean of these most likely ones is considered as *aposteriori* estimate.

The main difference between the proposed algorithm and the CPF algorithm is the selection process used by both the methods. Contrary to CPF algorithm, proposed algorithm uses some deterministic method for selection of particles for next time step. The selection is $(\mu+\lambda)$ selection of ES, which selects best μ points out of the total of $(\mu+\lambda)$ samples and supports rather than a (μ,λ) selection in which selection is made from λ supports, excluding μ points(Beyer and Schwefel, 2002). Determinism is due to the use of known probability distribution function (pdf) to create new population used in next time step. This approach, is analogous to multiple model adaptive estimation (MMAE) concept(Hanlon and Maybeck, 2000) and is particularly useful in real time system simulation.

4 TRACKING PROBLEM

4.1 System Used

Performances of two estimation algorithms, CPF and REPF is compared by designing these algorithms for tracking problem. A point moving with constant velocity in two dimensional plane is considered, where its distance and elevation angle are the measurements (as in case of radar measurement). Relating to Equation 3 and 4 above, $x_k = [x1\, x2\, \dot{x1}\, \dot{x2}]$, the discretized F matrix and measurement z are as follows:

$$F = \begin{bmatrix} 1 & 0 & \Delta k & 0 \\ 0 & 1 & 0 & \Delta k \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and

$$z = \begin{bmatrix} tan^{-1}((x(2)/x(1)) \\ sqrt(x(1)^2 + x(2)^2) \end{bmatrix} + v$$

where, x and $\dot{x}$ denotes the x position and velocity respectively. Clearly, this measurement model is non-linear (Li and Jilkov, 2003; Li and Jilkov, 2001) and thus, conventional KF is not applied. To simulate improper system dynamics, Δk in F is altered by incorporating $normrnd(0,0.1)$ error, which in turn, produces 8% error in X-position and Y-position. Practical significance of this may be explained by the fact that data may arrive at irregular interval which makes the system non-linear. For the purpose of simulation, it is assumed that, the object is initialized with the position [10 m,10 m] in X and Y axis respectively, and, initial velocity is assumed to be 1 m /sec along both the axes. Acceleration is assumed normally distributed and is modelled as white Gaussian noise, w_k. Parameters corresponding to variables used in algorithms used in this simulation are listed in Table 1.

Table 1: Parameters used in simulation of tracking problem.

Symbol	Value
Q	$(1e-6)/2.*eye(4)$
w	$N([0000]', sqrt(Q))$
R	$0.05/2.*eye(2)$
v	$N([0\ 0]', sqrt(R))$
X_0	$[10\ 10\ 1\ 1]'$
dx	$[0.1\ 0.1\ 0.1\ 0.1]'$
x_0	$X_0 + dx.*randn(4,1000)$

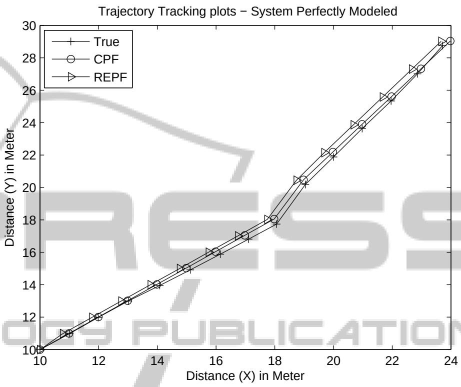

Figure 2: Tracking when system knowledge is proper.

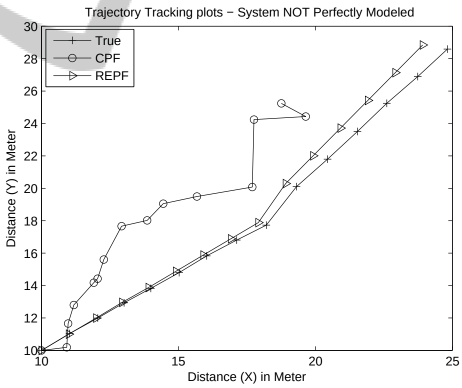

Figure 3: Tracking fails when system knowledge is not proper.

4.2 Result Analysis

Performance of CPF is compared with REPF. From Figure 2 it can be inferred that, the proposed algorithm works at par with CPF when the system is properly modelled. It is shown by simulation that, proposed algorithm manages to track satisfactorily in situation when the system is not perfectly modelled, while the CPF fails (see Figure 3). Proposed algorithm simulates multiple models by generating support points for every particles, and thus, manages to track the system when it is improperly modelled. Fig-

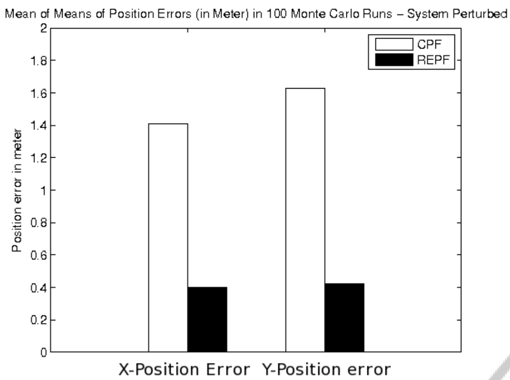

Figure 4: Mean of Means of Position Errors (in Meter) in 100 Monte Carlo Runs for both filters

ure 4 shows the result of 100 Monte Carlo runs. Mean of Means of position estimation has been considered for relative comparison of performances of two filters. Closer the value of this quantity to zero, better the performance of the filter. As described before, the value of Δk has been contaminated with around 8% error, which results in wrong estimation of X and Y position for CPF in wrongly modelled system, where REPF performs satisfactorily.

5 SOLUTION FOR TA PROBLEM

5.1 TA System Model

To capture complete non-linearity, complex navigation error equations can be used. Inclusion of gyro, accelerometer error components as system state in the state vector increases the overall complexity of computation. For practical implementation purpose, less complex error equations can be used to estimate misalignment angle accurately. Current work assumes the system state vector to be $x_k = [\delta\alpha\ \delta\beta\ \delta\gamma\ \delta v_n\ \delta v_e\ \delta v_d\ \delta L\ \delta h]$, where, $[\delta\alpha\ \delta\beta\ \delta\gamma]$ are attitude errors along N-E-D axes, $[\delta v_n\ \delta v_e\ \delta v_d]$ are velocity errors and $[\delta L\ \delta h]$ are errors in latitude and heights as described in (Titterton and Weston, 2004). Longitude as state variable has been omitted in current formulation because of its negligible contribution in estimation equations. State transition matrix is formulated based on the error propagation equations as described below.

$$\dot{\psi} = -\omega_{in}^n \times \psi - C_b^n \delta\omega_{ib}^b + \delta\omega_{in}^n \tag{6}$$

where, $\psi = \begin{bmatrix} \delta\alpha & \delta\beta & \delta\gamma \end{bmatrix}^T$ the alignment error vector,
$\delta\omega_{ib}^b = (\tilde{\omega}_{ib}^b - \omega_{ib}^b)$ the gyroscopic measurement errors in the slave system,
$\delta\omega_{in}^n = (\tilde{\omega}_{in}^n - \omega_{in}^n)$ the errors in the reference frame rate estimates.

$$\dot{v}^n = C_b^n f^b - g \tag{7}$$

where, v^n is the velocity of the mother in navigation frame, f^b is the output of accelerometers of daughter in body frame, along body axes and g is the gravity vector. C_b^n is the rotation matrix from body frame to navigation frame. Then the estimates of velocity error in the launcher (δv) will be –

$$\dot{\delta v} = f^n \times \psi + C_b^n \delta f^b \tag{8}$$

f^n is the accelerometer measurement of the master in body frame resolved in the navigation frame, and δf^b is the accelerometer noise.

In Equation 6 and Equation 8, C_b^n represents the Direction Cosine Matrix (DCM) which express the orientation of the daughter instrument cluster with respect to navigation frame (Titterton and Weston, 2004). For large initial misalignment, small angle approximation in constituent elements of this DCM is not possible, which, in turn, makes the system nonlinear. Referring to Figure 1, in absence of external aiding (or otherwise as well), mother measurement becomes erroneous, which in turn, corrupt the externally supplied state transition matrix. In such situation, particles of a CPF propagates through the erroneous state transition matrix and thus fails to perform satisfactorily. Better approach is to propagate particles through various state transition matrices and generate support points for each particle, which is essentially the practice followed in ES based PF.

Generation of multiple support points avoids resampling technique and still produces good estimate. Since the particles cover a larger space, the surviving points cover a larger space than CPF where the closest points are retained. Performance of REPF does not noticeably improve with respect to the performance of conventional PFs, when the knowledge of the system dynamics is accurate. However, the REPF can cover more non-linearity associated with the system and yields more realistic convergence when knowledge about the system is poor due to the ability of covering wider range of solution space in terms of generated support vectors for each particle which is not possible in case of CPF.

5.2 Performance Analysis

The current work implements and evaluates performances of CPF (with $N_s = 1000$ particles) and REPF (with $N_s = 1000$ particles and $n = 15$ support points for each particle). A regulator has been designed to correct the misalignment computationally. Performance of these filters have been evaluated in two different simulations by using Monte Carlo technique.

Table 2: Parameters used in simulation runs.

Parameters	Value
Initial Mis-alignment	$\delta\alpha = 5^o$ $\delta\beta = 8^o$ $\delta\gamma = 10^o$
Gyro Drift	$3^o/hr$
Accelerometer Bias	$500\mu g$
System noise covariance $Q(8\times 8)$	$diag([(4.7596e-7)\ (1.2185e-7)\ (1.9039e-7)\ (0)_{1\times 5}])$
Measurement noise covariance $R(5\times 5)$	$diag([(2.25e-4)_{3\times 1}\ (6.4e-4)\ (7.284e-4)]$

One simulation portrays the situation where system dynamics is known. In this simulation, state transition matrix has been kept unperturbed, and, all three algorithms are executed. In another simulation, state transition matrix has been made erroneous by injecting normally distributed noise to constituent elements of F matrix (zero mean with 10% variation), to capture the situation of improper system dynamics. Particles of REPF are propagated apriori through fifteen such matrices, and its performance is evaluated with that of CPF with 1000 particles in which particles are propagated through one state transition matrix. Parameters used in the simulations are given in Table 2.

For proof of concept, trajectory data has been generated from a simulated manoeuvre. Figure 5 shows profile of acceleration in North East Down (NED) directions used to generate the manoeuvre, based on the kinematic approach (Adhikari et al., 2002). It may be observed that, small amount of acceleration has been used in all three directions, which is desirable to avoid easy detection. Typical plots of misalignment angles $(\delta\alpha, \delta\beta, \delta\gamma)$ of CPF with 1000 particles and CPF with 3000 particles and REPF are given in Figures 6, 7, 8. All three filters take approximately same time to converge and the accuracy is also comparable. Figure 9 and 10 depicts the RMSE plots (Mean and STDs)

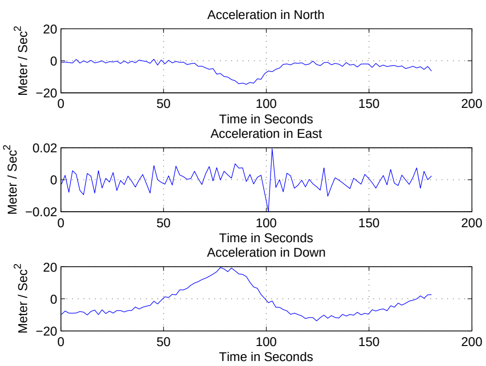

Figure 5: Acceleration profile used.

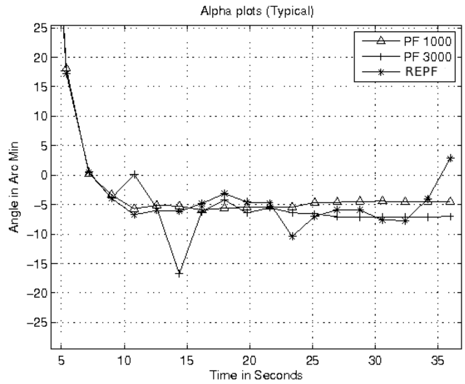

Figure 6: Typical Plot of $\delta\alpha$ (Zoomed).

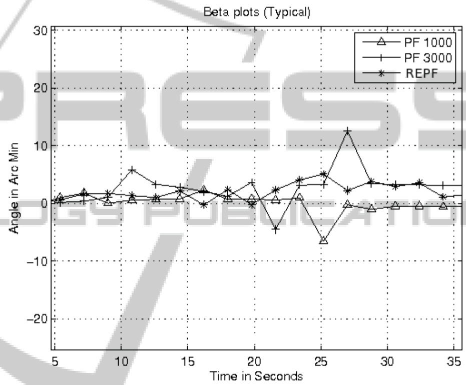

Figure 7: Typical Plot of $\delta\beta$ (Zoomed).

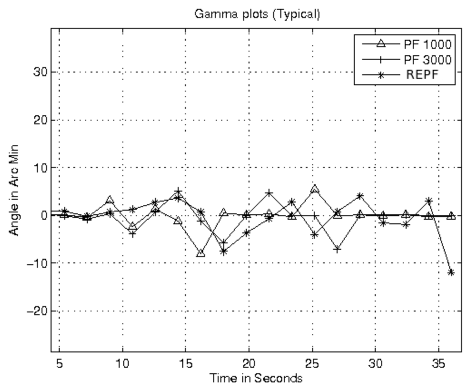

Figure 8: Typical Plot of $\delta\gamma$ (zoomed).

of misalignment angles for 100 runs of Monte Carlo simulation (MCS) which shows the similar nature of convergence of all three algorithms. Comparisons of mean of means of the RMSE values of the misalignment angle estimates computed over 100 MCS runs are tabulated in Table 3. Due to presence of regulator, the expected value is zero. The values depicted is a clear indicator of the comparable performance of above three filters.

Next, MCS is performed for the scenario where

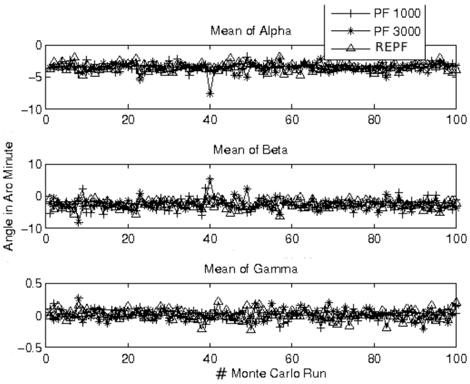

Figure 9: Mean of δα,δβ,δγ for 100 MC runs.

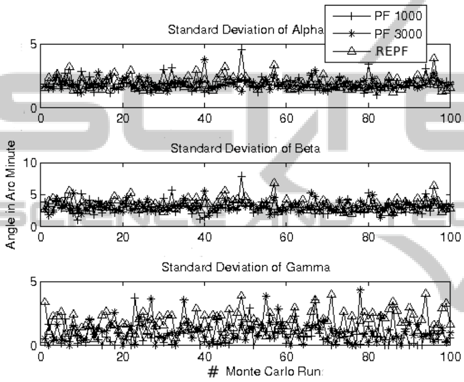

Figure 10: Variance of δα,δβ,δγ for 100 MC runs.

Table 3: Performance of three filters for 100 MC runs when the system knowledge is correct.

Mean of means (*in arc min*)	CPF with 1000 particles	CPF with 3000 particles	REPF
δα	-3.372067	-3.542256	-3.463119
δβ	-2.749540	-2.190724	-2.577882
δγ	-0.022702	0.011495	0.001139

knowledge of the system is improper. While the conventional filters fail to converge, REPF delivers good convergence. Values of mean of means of the CPF and REPF algorithm for 100 MCS runs in this scenario has been tabulated in Table 4. It clearly indicates the inefficiency of CPF algorithm in overcoming lack of knowledge of system dynamics. In this respect, the proposed REPF algorithm performs well.

5.3 Complexity Analysis

Total complexity of running the TA algorithm in each iteration can be expressed as the sum of:

Table 4: Performance of CPF and REPF over 100 MC runs when system knowledge is incorrect.

Mean of means (*in arc min*)	CPF with 1000 particles	REPF
δα	36.709314	-3.384981
δβ	-100.142260	-2.591800
δγ	27.838801	0.024971

- Time to generate *apriori* estimates (uses common mother INS data)
- Time to select the appropriate support vector (uses daughter INS measurement)
- Time to arrive at *aposteriori* estimate (daughter specific computation)

In the first step, the generation of *apriori* estimate of the support particles is linear in number of particles but would need amortized analysis that depends on the size of state vector. Generation of each *apriori* estimate (support points for all particles) requires multiplication of matrices having size of the state vector. The size of state vector is also responsible for the space complexity to be handled by the algorithm during execution. Any reduction in run-time memory requirement in turn can increase the processor efficiency in handling floating point operations.

The current work restricts the number of state variables to 8. There are TA algorithms which additionally estimate state variables like gyro and accelerometer bias, which increases the number of state variables twofold or even more (Kong and Nebot, 1999), so that the amortized value increases eightfold or more. The bias estimation as state variables may result in more accurate state estimation of misalignment angles. But the present work tries to improve performance even with inaccurate system model, where estimation of bias cannot improve the accuracy. Hence the overhead of these extra states has been eliminated in this work, resulting in better amortized complexity.

The second step involves particle sampling. A SIR-PF with N number of particles, employs a systematic re-sampling technique which runs in $O(N)$ time, which dominate the overall time complexity of the algorithm (Arulampalam et al., 2002; Carpenter et al., 1999). REPF does not require any such re-sampling, and, performs a sorting algorithm to choose the best of n support points for each N_s particles. Such sorting requires $O(n^2)$ time in the worst case, yielding an overall time complexity for REPF algorithm $T(N)$ as described below:

$$T(N) = N_s\, O(n^2) \cong c\, O(N_s) \tag{9}$$

where, $c = n^2$. As $n \ll N_s$, it can be conferred that, the execution time of REPF is comparable to that of a SIR-PF. The computation of *aposteriori* estimate is linear in N_s, so that the overall complexity also stays linear.

6 CONCLUDING REMARKS

The work addresses the issues arising out of the lack of knowledge about time varying state transition matrix that is used for system modelling. A particle filter based algorithm based on evolutionary strategy has been designed to tackle such scenario. Performance of the designed filter algorithm has been compared with CPF by employing Monte Carlo simulation. Robustness of filter performance has been studied by applying to tracking problem. Then the similar algorithm is applied to handle the problem of non-linearity in TA problem in presence of large initial misalignment angle as well as poorly modelled system.

Their performance is comparable when the nonlinearity of the system is well configured. But in situations where knowledge about the system is poor, REPF performs better than that of conventional PF, as evident from large number of Monte Carlo runs. Time and space complexity associated with the real time implementation of such filter is discussed in details. It is shown that the complexity is comparable and amortized analysis shows improvement in overall complexity.

In the case of multiple daughter ejection, scope of multi-threading of TA algorithms and faster and concurrent convergence of TA algorithms needed for multiple daughter ejection have been discussed in (Das et al., 2009). It was identified that, some task involving mother INS data, that is common for all daughters, can be assigned to the mother On Board Computer (OBC), thereby reducing computational overhead of daughter OBC. This concept will be useful in the REPF algorithm implementation. The computation of state transition matrix and propagation of particles for each daughter needs only mother INS data which may be run as different threads in the mother OBC and can be passed to daughter as and when required. This reduces daughter processor time overhead, which may be utilized solely for daughter specific tasks as discussed above. This can help in simultaneous convergence of TA algorithms of all the daughters.

ACKNOWLEDGEMENTS

The work is partially supported from DRDO project. Inputs from the scientists M. Kumar, S. Das and Dr. S.K Chaudhuri were valuable. Shrutilekha Santra contributed as project fellow.

REFERENCES

Adhikari, P., Mukherjee, A., and Nandi, P. (2002). Sensor data simulation for studies of navigation system: A kinematic approach. *Conference*, pages 1–12.

Ali, J. and Jiancheng, F. (2004). Alignment of strapdown inertial navigation system : A literature survey spanned over the last 14 years. *Science*, 100083:1–12.

Arulampalam, M. S., Maskell, S., Gordon, N., and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188.

Bemporad, A., Morari, M., Dua, V., and Pistikopoulos, E. N. (2002). The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20.

Beyer, H.-G. and Schwefel, H.-P. (2002). Evolution strategies: A comprehensive introduction. 1(1):3–52.

Carpenter, J., Clifford, P., and Fearnhead, P. (1999). Improved particle filter for nonlinear problems. *IEE Proceedings - Radar, Sonar and Navigation*, 146(1):2.

Chattaraj, S., Mukherjee, A., and Chaudhuri, S. K. (2013). Transfer alignment problem: Algorithms and design issues. *Gyroscopy and Navigation*, 4:130–146.

Das, A., Chakraborty, S., Mukherjee, A., and Chaudhuri, S. (2009). A critical study on using multithreading for transfer alignment of precision guided daughter vehicles. In *National Conference on Modeling and Simulation (NCMS 2009) held at DIAT (DU), Pune during Dec, 2009.*

Ding, G., Zhou, W., Hao, Y., and Cui, G. (2009). Deterministic particle filter based transfer alignment method for large misalignment angle. In *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, pages 3387–3392.

Dmitriyev, S., Stepanov, O., and Shepel, S. (1997). Nonlinear filtering methods application in INS alignment. *Aerospace and Electronic Systems, IEEE Transactions on*, 33(1):260–272.

Groves, P. (1999). Transfer alignment using an integrated INS/GPS as the reference. *Journal of Navigation*, (2):731–737.

Hanlon, P. and Maybeck, P. (2000). Multiple-model adaptive estimation using a residual correlation kalman filter bank.

Julier, S. and Uhlmann, J. (1997). A new extension of the Kalman filter to nonlinear systems. *Int Symp AerospaceDefense Sensing Simul and Controls*, 3:182–193.

Kong, X. and Nebot, E. (1999). Development of a nonlinear psi angle model for large misalignment errors and its application in ins alignment and calibration. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 2, pages 1430–1435.

Kwok, N., Fang, G., and Zhou, W. (2005). Evolutionary particle filter: re-sampling from the genetic algorithm perspective. In *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 2935–2940.

Li, X. and Jilkov, V. (2003). Survey of maneuvering target tracking. part i. dynamic models. *Aerospace and Electronic Systems, IEEE Transactions on*, 39(4):1333–1364.

Li, X. R. and Jilkov, V. P. (2001). A survey of maneuvering target tracking - part iii: Measurement models. pages 423–446.

Titterton, D. H. and Weston, J. L. (2004). *Strapdown Inertial Navigation Technology*, volume 17. Peter Peregrinus Ltd.

Uosaki, K., Kimura, Y., and Hatanaka, T. (2003). Nonlinear state estimation by evolution strategies based particle filters. In *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on*, volume 3, pages 2102–2109 Vol.3.

Uosaki, K., Kimura, Y., and Hatanaka, T. (2004). Evolution strategies based particle filters for state and parameter estimation on nonlinear models. In *Evolutionary Computation, 2004. CEC2004. Congress on*, volume 1, pages 884–890 Vol.1.

Wan, E. A. and Van Der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. *Technology*, v(3):153–158.